RMK EMU DOC 2

MANUAL DEL EMULADOR DEL RMK85 VERSION TERMINAL

PUB. NO: 0281085 REV: A.

MANUAL DEL USUARIO DEL NIVEL 3

Pub. Nro. 0281085-Rev. A

CARACTERISTICAS TECNICAS DEL NIVEL 3

- Emula las señales del microprocesador 8085 en tiempo real.
- Trabaja con los 64K bytes del mapa de memoria de la CPU emulada.
- Trabaja con las 256 direcciones del mapa de I/O de la CPU emulada.
- Permite partir el mapa de memoria de la CPU emulada en zonas de 1K byte a partir de la direccion 0000h.
- Permite transferir zonas desde la direccion 0000h del mapa de memoria de la CPU emulada, a la direccion 0000h del mapa del emulador.
- Todos los comandos del Monitor del RMK/85 pueden usarse sobre la plaqueta emulada.
- Funciona en version Hexadecimal o en version Terminal.
- Realiza el disassembler de cualquier zona de memoria (version terminal).
- SINGLE-CYCLE DISPLAY BOARD incluido que:
    - permite poner a la CPU en estado de WAIT.
    - permite colocar break-points por hardware en cualquier direccion de memoria o de I/O.
    - permite observar el estado (en binario) de las lineas de address, dato y status de la CPU.
- En la version terminal se pueden examinar y alterar el estado de los flags, en binario, uno por uno.
- Los comandos Examine y Trace (version terminal) muestran el disassembler del contenido del PC.
- Se conecta directamente al bus de la plaqueta de aplicacion PCI-85.
- El tamaño de la plaqueta y conexionado es compatible con el Sistema RMK/85.

MANUAL DE INSTALACION DEL NIVEL 3 DEL RMK/85

Fuente de alimentacion

Se debe conectar al Nivel 3, la fuente de alimentacion, a traves del conector J3. Los requisitos y conexionados de esta fuente de alimentacion son identicos a los del Nivel 1 (ver Manual del Nivel 1 del RMK/85). El consumo aproximado es de 900mA.

Conexionado a los niveles 1 y 2

La plaqueta del Nivel 3 del RMK/85 se conecta al bus del RMK/85 a traves del conector J1. La conexion se realiza por medio de un arnes de cinta plana, tipo DAISY-CHAIN polarizado de 64 vias.

Nota importante: es conveniente utilizar un solo Nivel 2 cuando se trabaja con el emulador para obtener una mayor confiabilidad en el sistema.

Conexionado al SINGLE-CYCLE DISPLAY BOARD

El SINGLE-CYCLE DISPLAY BOARD se conecta por medio de un arnes polarizado de cinta plana de 40 vias. al conector J4 del Nivel 3.

Conexionado a la plaqueta de aplicacion PCI-85

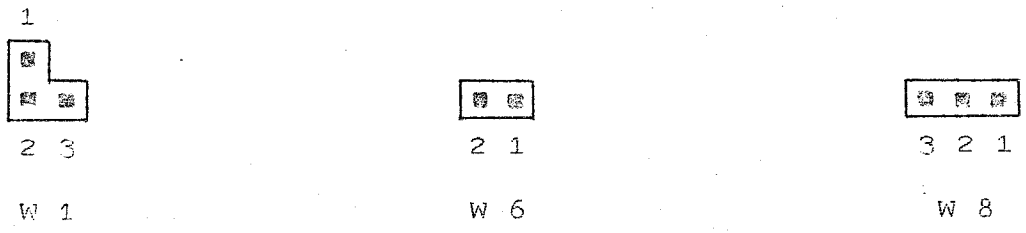
La plaqueta de aplicacion PCI-85 se conecta a traves de su conector J1. al conector J2 del Nivel 3. por medio de un arnes de cinta plana de 50 vias: en este conector estan todas las señales del microprocesador 8085.

Nota importante: cuando se conecta la plaqueta de aplicacion PCI-85 al Nivel 3. se debe quitar el microprocesador de la misma, puesto que todas las señales del 8085 son enviadas desde el emulador.

MANUAL DEL USUARIO DEL NIVEL 3

Jumpers del Nivel 3

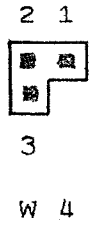
Jumpers para Seleccion de Modo del Single-Cycle



W1		W6	W8		
1-2	2-3	1-2	1-2	2-3	
*		*		*	Se detiene en cualquier lugar dentro o fuera del Monitor.
	*	*	*		Se detiene solamente en el programa del usuario despues de dar Go o Trace.
	*		*		Se detiene cuando lee o escribe el emulando dentro o fuera del programa del usuario. Precaucion: En este modo las interrupciones no deben estar presentes (RST 5.5, 6.5, 7.5, INTR, HOLD, TRAP).

* Signifca pines unidos.

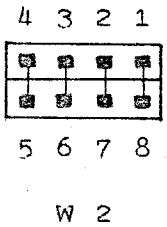
Jumper para habilitacion de la entrada serie (SID)



SID a la CPU. Debe estar 1-2 unidos mientras esta puesto el puente de la interfase a grabador en el Nivel 1.

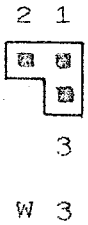
MANUAL DEL USUARIO DEL NIVEL 3

Jumpers para señales de control (pull up's)



Pull up's sobre las señales de $\overline{RD}$. $\overline{WR}$. $\overline{INTA}$, IO/$\overline{MEM}$ que van a la plaqueta emulada. Generalmente estan colocados.

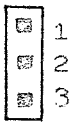
Jumper para la interrupcion TRAP



W3		
1-2	1-3	
*		TRAP proveniente de la plaqueta emulada. habilitado.
	*	TRAP proveniente de la plaqueta emulada. inhabilitado.

* Significe pines unidos.

Jumper de seleccion logica de ALE



W 5

W5		
1-2	2-3	
*		ALE al conector J2 (utilizar de esta manera).
	*	$\overline{ALE}$ al conector J2.

* Significa pines unidos.

Jumper de habilitacion del Single Cycle

W 7

Single Cycle Off cuando esta cerrado (idem llave ON/OFF sobre frente).

MANUAL DEL MONITOR EMULADOR VERSION TERMINAL

El Monitor del Emulador del RMK/85 tiene 3 modos de funcionamiento:

1- Monitor comun.
2- Monitor Emulador.
3- Monitor Emulador con move externo automatico.

Al encender el equipo, toma control de la CPU el programa Monitor, el cual detecta si esta conectada o no la plaqueta del Nivel 3. Si esta no lo esta, solamente se podran ejecutar los comandos de la version Monitor Terminal comun.

Cuando el Monitor termina con dicha deteccion y la inicializacion del equipo, se observa en la pantalla el mensaje de presentacion del mismo:

```
Programa Monitor RMK/85
Vers 3.2
M Ok
```

La ultima linea indica que el monitor se encuentra a la espera de un comando por parte del usuario.

Los combndos para los 3 modos del Monitor del Emulador se pueden dividir en:

1) Comandos iguales al Monitor del RMK/85.
2) Comandos modificados.
3) Comandos nuevos del Monitor Emulador.

MANUAL DEL USUARIO DEL NIVEL 3

1) Comandos iguales al Monitor del RMK/85.

Alterar A <buffer low>,<buffer high>,<prom low>
Ingreso de parametros para ser usados en todo comando del Prom Programmer.

Basic B
Ir al BASIC del RMK/85.

Comparar C
Compara el contenido del buffer con el contenido de la Prom.

Editor E
Ir al Editor/Assembler del RMK/85.

Fill F <low address>,<high address>,<data>
Llena la zona de memoria dada por los limites con un dato.

Go G <PC>,<BP>
Ejecutar programa de usuario.

Help H
Listado de comandos validos del RMK/85.

Input I <I/O address>
Lee la puerta de I/O mostrando su valor.

Transferir K
Transfiere el contenido de la Prom al buffer del Prom Programmer.

Listar L
Lista el contenido de la Prom.

Move M <low address>,<high address>,<dest. low>
Transfiere el bloque de memoria entre los parametros low y high address, a la zona a partir de <dest. low>.

Output O <I/O address>,<data>
Se escribe la puerta de direccion <I/O address>, con el dato <data>.

Programar P
Programar la Prom, previa verificacion de borrado.

Programar s/v Q
Programar la Prom. sin verificacion de borrado.

Read R <low address>,<numero>
Lee un archivo de cassette a la zona de memoria a partir de <low address>.

Sustituir S <address>
Examina y/o modifica el contenido de posiciones de memoria consecutivas a partir de <address>.

Listar Buffer U
Lista el contenido del buffer del Prom Programmer.

Ver parametros V
Muestra los parametros actuales del Prom Programmer.

Write W <low address>,<high address>.<numero>
Escribe en cassette la porcion de memoria entre los limites low y high address.

Borrado Z
Efectua una verificacion de borrado de la Prom.

Printer cntl-P
Habilita o deshabilita el Printer.

El empleo y modo de ejecucion de estos comandos estan descriptos en el manual del Monitor Terminal del RMK/85.

Nota: Cuando no esta conectada la Expansion del Monitor (Nivel 3) los comandos Examine (X) y Trace (T) se ejecutan de la misma manera que estan descriptos en el manual Monitor Terminal.

2) Comandos Modificados.

```
*****************************************************************
*                                                               *
*          D     <low address>,<high address>                   *
*                                                               *
*          Display                                              *
*****************************************************************
```

Se muestra el contenido hexadecimal y ASCII (si es imprimible), de la zona de memoria especificada. El listado se puede detener momentaneamente con la tecla 'S' (start/stop), y continuar con cualquier otra tecla, o bien terminar con cualquier otra tecla. Si se omite el <high address>, se listan 64 posiciones de memoria a partir de <low address>. Si se omiten ambos parametros se listan 64 posiciones a partir del listado anterior. De esta manera se pueden listar porciones consecutivas de la memoria en bloques de 64 bytes.

Ejemplo:

```
M Ok D10.14[cr]
0010 52 05 3D 5F 74  R.=_t

M Ok D10[cr]
0010 52 05 3D 5F 74 66 77 77  R.=_tfww
0018 00 01 02 03 04 05 06 07  ........
0020 00 01 02 03 04 05 06 07  ........
0030 00 01 02 03 04 05 06 07  ........
0038 00 01 02 03 04 05 06 07  ........
0040 00 01 02 03 04 05 06 07  ........
0048 00 01 02 03 04 05 06 07  ........
M Ok D[cr]
0050 00 01 02 03 04 05 06 07  ........
0058 00 01 02 03 04 05 06 07  ........
0060 00 01 02 03 04 05 06 07  ........
0068 00 01 02 03 04 05 06 07  ........
0070 00 01 02 03 04 05 06 07  ........
0078 00 01 02 03 04 05 06 07  ........
0080 00 01 02 03 04 05 06 07  ........
0088 00 01 02 03 04 05 06 07  ........
```

```
*****************************************************************
*                                                               *
*          T     <cantidad>,<PC>                                *
*                                                               *
*          Trace                                                *
*****************************************************************
```

Se ejecuta, paso a paso, el programa de usuario a partir de la direccion dada por <PC>: si se omite este parametro, se ejecuta a partir de la direccion actual del Program Counter de la CPU. La cantidad de pasos a ejecutar esta dada por el parametro <cantidad>: si se omite este parametro se ejecuta un solo paso de programa. La ejecucion puede detenerse momentaneamente con la tecla 'S', y continuar luego con cualquier tecla, o bien terminar con cualquier otra tecla antes de cumplir la cantidad especificada de pasos de programa. Luego de cada paso de programa, se imprime el contenido de todos los registros de la CPU, y el mnemonico del contenido de la direccion a que apunta PC, de identica manera al comando 'X'.

Ejemplo:

M Ok T[cr]

AC	SZAPC	BC	DE	HL	M	SP	PC	(PC)	MNEMONICO
07	10111	F812	F8F0	2F7C	7C	F8E0	0001	23	INX H

Se ejecuta un paso del programa de usuario a partir del Program Counter actual.

M Ok T04[cr]

AC	SZAPC	BC	DE	HL	M	SP	PC	(PC)	MNEMONICO
07	10111	F812	F8F0	2F7D	7D	F8E0	0002	23	INX H
07	10111	F812	F8F0	2F7E	7E	F8E0	0003	23	INX H
07	10111	F812	F8F0	2F7F	7F	F8E0	0004	23	INX H
07	10111	F812	F8F0	2F80	B8	F8E0	0005	23	INX H

Se ejecutan 4 pasos del programa de usuario a partir del Program Counter actual.

M Ok T03.0100[cr]

AC	SZAPC	BC	DE	HL	M	SP	PC	(PC)	MNEMONICO
07	10111	F811	F8F0	2F80	B8	F8E0	0101	0B	DCX B
07	10111	F810	F8F0	2F80	B8	F8E0	0102	0B	DCX B
07	10111	F80F	F8F0	2F80	B8	F8E0	0103	0B	DCX B

Se ejecutan 3 pasos del programa de usuario a partir de la direccion 0100h.

```
*****************************************************************
*                                                               *
*          X     <registro>                                     *
*                                                               *
*          Examine                                              *
*****************************************************************
```

Este comando permite examinar y/o alterar el contenido de los registros de la CPU. Estos son:

A		acumulador	-8 bits
	S	flag de signo	-1 bit
	Z	flag de zero	-1 bit
F	A	flag de aux carry	-1 bit
	P	flag de parity	-1 bit
	C	flag de carry	-1 bit
B		par BC	-16 bits
D		par DE	-16 bits
H		par HL	-16 bits
S		Stack Pointer	-16 bits
P		Program Counter	-16 bits

En el listado de registros tambien aparecen:

M	contenido de (HL)	-8 bits
(PC)	contenido de (PC)	-8 bits

MNEMONICO disassembler correspondiente al contenido de (PC)

M y (PC) no pueden ser modificados pues no son registros propios de la CPU.

La modificacion del contenido de los registros es de a uno, especificando el nombre del registro, o secuencial separando cada uno con un espacio o una coma.

Para modificar el estado de los flags se debe ingresar, despues de tipear la letra 'F', la letra correspondiente al flag elegido, o bien teclear espacio o coma y se va modificando secuencialmente el estado del flag de signo (S), zero (Z), auxiliary carry (A), parity (P) y carry (C).

Si se continua con un separador valido (espacio o coma), luego de haber mostrado el estado del flag de carry, el comando 'X' continua con los estados de los registros BC, DE, HL, etc.

Ejemplo:

M Ok X[cr]

AC	SZAPC	BC	DE	HL	M	SP	PC	(PC)	MNEMONICO
02	00010	0411	0405	C400	38	F8E0	0000	12	STAX D

De esta manera se presenta el contenido de todos los registros, M, (PC) y el disassembler correspondiente al (PC).

Ej. para modificar BC:

M Ok XB 0411-3455[cr]

De esta manera modificamos el contenido del par de registros BC.

Anterior	Actual
BC=0411	BC=3455

Ej. para modificar flags, BC y DE:

M Ok XFP 1-0 C 0-1 3455-1234 0405-1111[cr]

De esta manera modificamos el contenido del flag de parity, el de carry y los pares de registros BC y DE consecutivamente.

Anterior	Actual
FP=1	FP=0
FC=0	FC=1
BC=3455	BC=1234
DE=0405	DE=1111

3) Comandos nuevos del Monitor Emulador.

```
*****************************************************************
*                                                               *
*          J     <modo>                                         *
*                                                               *
*          Seleccion del Modo del Monitor                       *
*****************************************************************
```

Este comando permite elegir el modo de funcionamiento.
Modo puede valer 0, 1 o 2.

J <0> : Monitor comun.
J <1> : Monitor Emulador.
J <2> : Monitor Emulador con move externo automatico
J [cr]: muestra el modo y parametros.

Cuando se escoge el modo J0, aparece en la pantalla la leyenda :

M Ok

Esto le indica al usuario que esta trabajando con el Monitor comun version terminal.

Cuando se escoge el modo J1, aparece en la pantalla el mensaje:

Ingrese direccion zona RAM en emulando: <address>-

El programa monitor queda a la espera del ingreso de la direccion de RAMEMU por parte del usuario: <address> es el valor actual de RAMEMU.
Luego del ingreso de RAMEMU, aparece en pantalla el mensaje:

Ingrese el limite mapa de memoria: <MSB address>-

El programa monitor queda a la espera del ingreso de la direccion del limite que divide al mapa de memoria en dos partes: la parte inferior para el emulador y la parte superior para el emulado.
Este limite se ingresa colocando el byte mas significativo de dicha direccion y tiene que estar en bloques de 1K byte, por ejemplo: son validos los limites 00, 04, 08, 0C, 10, 14, etc.
<MSB address> es el byte mas significativo de la direccion actual del limite del mapa de memoria.

A continuacion aparece en la pantalla la leyenda:

E <limite> Ok

Esto le indica al usuario que se esta dentro del programa Monitor Emulador y a la espera de un comando; <limite> muestra el byte mas significativo de la direccion del limite del mapa de memoria.

RAMEMU: Esta es una direccion que debe pertenecer a un lugar de RAM dentro de la zona del mapa de memoria del equipo emulado, donde haya, por encima de esta direccion 4 bytes libres para que el programa del emulador guarde la direccion de retorno y dos instrucciones mas; y por debajo de la misma lugares libres para el STACK POINTER del usuario.

Cuando se escoge el modo J2, el programa monitor pide al usuario el ingreso de los mismos parametros que en el modo J1, con la diferencia de que al terminar de ingresar el limite del mapa de memoria, se realiza la transferencia a la parte del mapa de memoria que pertenece al emulador, de la zona de memoria del equipo emulado que quedo excluida del mapa de memoria por el limite que se ingreso (en el caso de que este no haya sido 00), permitiendo modificar el programa de la CPU emulada.

Cuando el programa monitor termina de realizar este MOVE EXTERNO, aparece en la pantalla la leyenda:

E <limite> Ok

Esto le indica al usuario que se esta dentro del programa Monitor Emulador y a la espera de un comando.

Cuando se tipea J [cr], el comando muestra en pantalla el modo en que se esta trabajando. Si se esta en modo Monitor comun muestra 00. Si se esta en alguno de los dos modos del Monitor Emulador, ademas de mostrar el nro. de modo (01 o 02), muestra la direccion de RAMEMU y la direccion del limite del mapa de memoria.

Ejemplo:

M Ok J1[cr]

Ingrese direccion zona RAM en emulando: 1000-08FC[cr]

Ingrese limite mapa memoria: 00-04[cr]

E 04 Ok

De esta manera entramos en el Modo Emulador. habiendo establecido la direccion de RAMEMU en la 08FCh y el limite del mapa de memoria en la direccion 0400h.

E 04 Ok J[cr]
01.08FC.0400

De esta manera obtenemos los datos del Modo actual, la direccion de RAMEMU, y la direccion del limite del mapa de memoria.

Nota importante: La direccion de RAMEMU debe estar siempre por encima del limite del mapa de memoria.

```
*************************************************************
*                                                           *
*          N    <cantidad bloques>                          *
*                                                           *
*          Move Externo                                     *
*************************************************************
```

Este comando transfiere el contenido de la memoria del dispositivo emulado a la memoria RAM del emulador, desde la direccion ØØØØ hexa, la cantidad de bolques especificados.

Un bloque equivale a 1ØØ bytes hexadecimales.

Se debe tener en cuenta que la memoria del emulador esta comprendida entre la direccion ØØØØh hasta la direccion del limite del mapa de memoria (exclusive).

Ejemplo:

E Ø4 Ok NØ4[cr]

De esta manera se transfieren 4 bloques de memoria (de 1ØØ bytes de extension cada uno) del dispositivo emulado, al mapa de memoria del emulador desde la direccion ØØØØh hasta la direccion Ø4ØØh (exclusive).

E 1Ø Ok NØ7[cr]

De esta manera se transfieren 7 bloques de la zona de memoria del dispositivo emulado desde la direccion ØØØØh, al mapa de memoria del emulador desde la direccion ØØØØh hasta la direccion Ø7ØØh (exclusive).

Nota importante: Este comando no se puede ejecutar desde el modo Monitor comun.

```
*****************************************************************
*                                                               *
*          Y    <low address>.<high address>                    *
*                                                               *
*          Disassembler                                         *
*****************************************************************
```

Se muestran las direcciones. contenidos de esas direcciones y el mnemonico correspondiente al set de instrucciones del 8085 de la zona especificada. El listado puede detenerse momentaneamente con la tecla 'S' (start/stop) y continuar con cualquier otra tecla. o bien terminar con cualquier otra tecla. Si se omite el <high address>. se listan 12 lineas a partir del <low address>. Si se omiten ambos parametros se listan 12 lineas a partir de la ultima direccion desensamblada anteriormente.

Ejemplo:

```
M Ok Y0.7[cr]
0000  00= NOP
0001  AA= XRA  D
0002  58= MOV  E.B
0003  59= MOV  E.C
0004  46= MOV  B.M
0005  B1= ORA  C
0006  C7= RST  0
0007  04= INR  B

M Ok Y[cr]
0008  09= DAD  B
0009  06= MVI  B,14
000B  CD= CALL 08FB
000E  21= LXI  H,F876
0011  11= LXI  D,0918
0014  06= MVI  B,0D
0016  CD= CALL 08FB
0019  21= LXI  H,0000
001C  22= SHLD F88D
001F  3E= MVI  A,40
0021  C3= JMP  0027
0024  C3= JMP  0079
```

SINGLE-CYCLE DISPLAY BOARD

Este es un dispositivo que permite seguir programas colocando Break-points por hardware en cualquier direccion de memoria o de I/O, o seguir el programa ciclo a ciclo de maquina (Op Code Fetch, Memory read, Memory write, I/O read, I/O write, $\overline{\text{INTA}}$ y BI).

Posee 18 llaves: una para habilitarlo o no (ON/OFF), una llave para seleccionar las tres formas de operacion: RUN, Break-point, o Single-Cycle, y 16 llaves para setear la direccion del Break-point.

Emplea 2 pulsadores, uno para la ejecucion de acuerdo con la forma de operacion elegida por la llave de modo, llamada RUN, y otra para detener la ejecucion llamada STOP, que pone a la CPU en estado de WAIT.

La seccion de display de informacion esta compuesta por 16 leds indicadores del contenido del bus de address, 8 leds que muestran el contenido del bus de datos, 6 leds para mostrar el bus de control y un led que indica si la direccion donde esta el programa pertenece al mapa del emulador o al del equipo emulado.

Modos de funcionamiento

Con la llave de modo en la posicion RUN, y oprimiendo el pulsador RUN, el programa apuntado por PC comienza a ejecutarse en forma continua. Para detener la ejecucion basta con pulsar la tecla de STOP o cambiar la llave de modo a la posicion de SINGLE-CYCLE.

Colocando la llave de modo en la posicion BKP (Break-point), oprimiendo el pulsador RUN y habiendo seteado las llaves de address en una determinada direccion de memoria o de I/O, el programa comienza a ejecutarse en forma continua desde la direccion apuntada por PC hasta que pase por la direccion seleccionada como Break-point, lugar donde se detiene.

Para colocar un break-point en una direccion de I/O se debe repetir dicha direccion en el byte mas significativo y menos significativo de las llaves de address, por ejemplo, si se desea colocar un break-point en el address de I/O Ø8 hexadecimal (ØØØØ1ØØØ binario) se deben poner las llaves apuntando al address Ø8Ø8 hexadecimal (ØØØØ1ØØØ ØØØØ1ØØØ binario).

Con la llave de modo en la posicion SC (SINGLE-CYCLE) cada vez que se oprime el pulsador RUN el programa ejecuta un ciclo de maquina de la instruccion apuntada por PC, mostrando en los leds los contenidos del bus de address, el de datos y el de control.

Ejemplo de debugging con el Emulador version Terminal

A continuacion se describe a modo de ejemplo la aplicacion del emulador para emular una plaqueta PCI-85 con una Eprom 2716.

Para comenzar, se debe emplear el comando J (seleccion de Modo del Monitor) para entrar por ejemplo, en el Modo Emulador con Move Externo automatico.

Una vez dentro de este modo el programa pide el ingreso de RAMEMU y del limite del mapa de memoria.

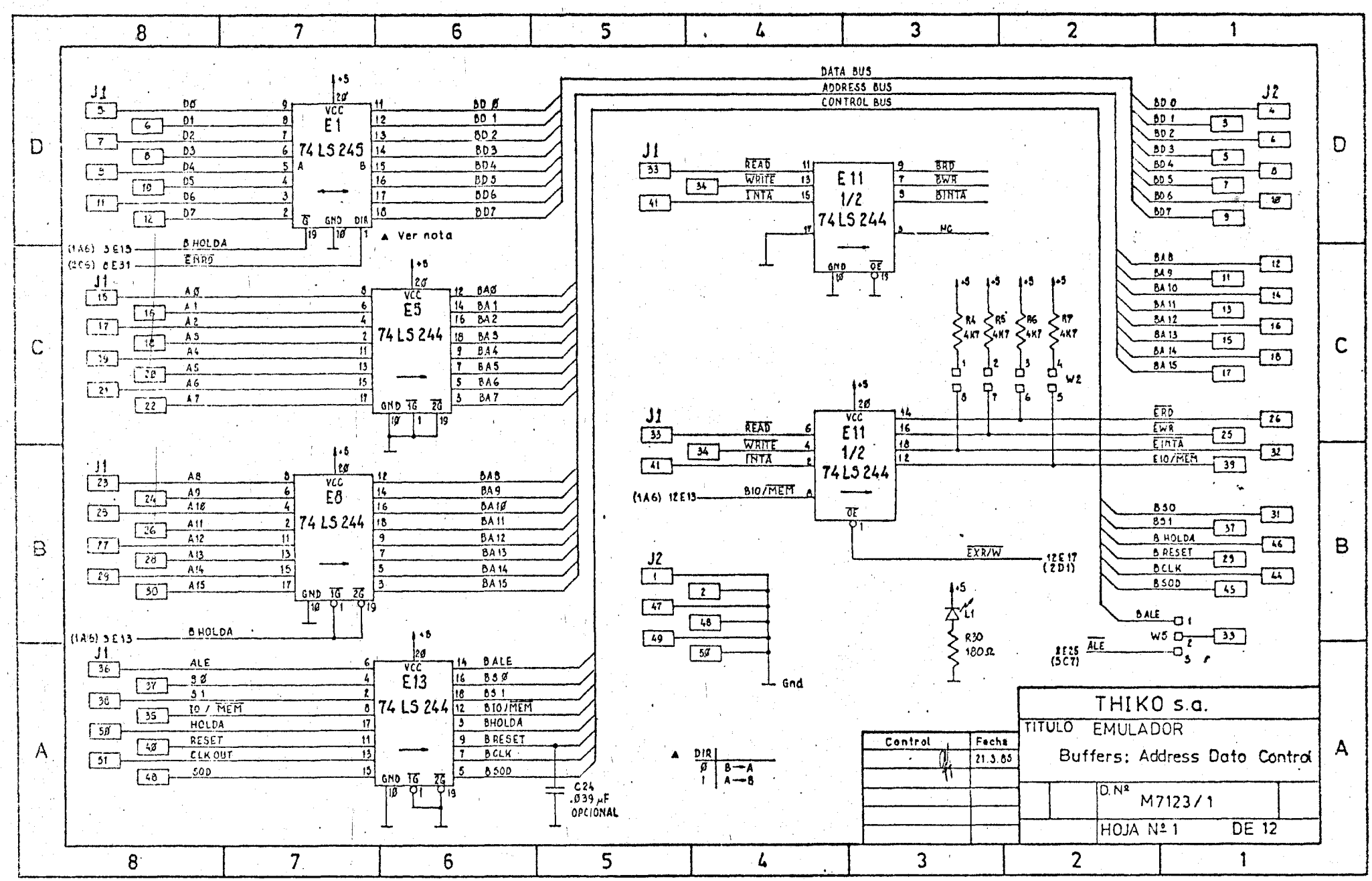
La PCI-85 con 2716 posee la RAM del RI/O desde la direccion 0800h hasta la 08FFh : la direccion de RAMEMU entonces, puede ser cualquiera dentro de estos limites y que el programa no use para guardar datos o variables, teniendo en cuenta que por debajo de RAMEMU quedara situado el nuevo Stack Pointer de usuario.

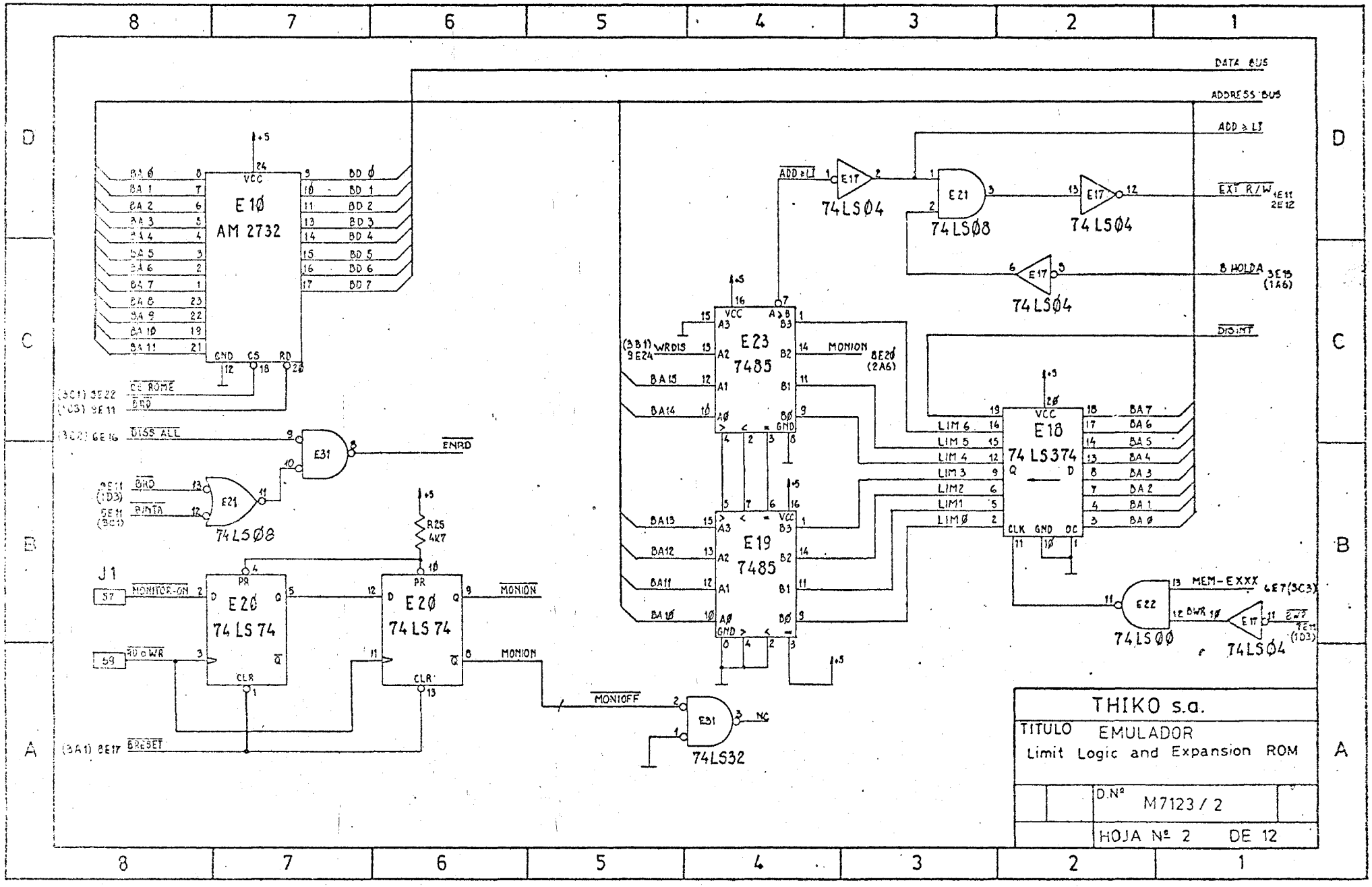
Al terminar de ingresar RAMEMU se debe ingresar el limite del mapa de memoria que lo divide en dos zonas: la parte baja para el emulador y la parte alta para el emulado. Como la PCI-85 posee, en este caso, una 2716 (de 2K byte), el limite que ingresaremos sera 08 (que es el byte mas significativo de la direccion 0800h), entonces el programa realiza el Move Externo colocando en la RAM del kit del emulador todo el programa existente en la ROM de la PCI-85.

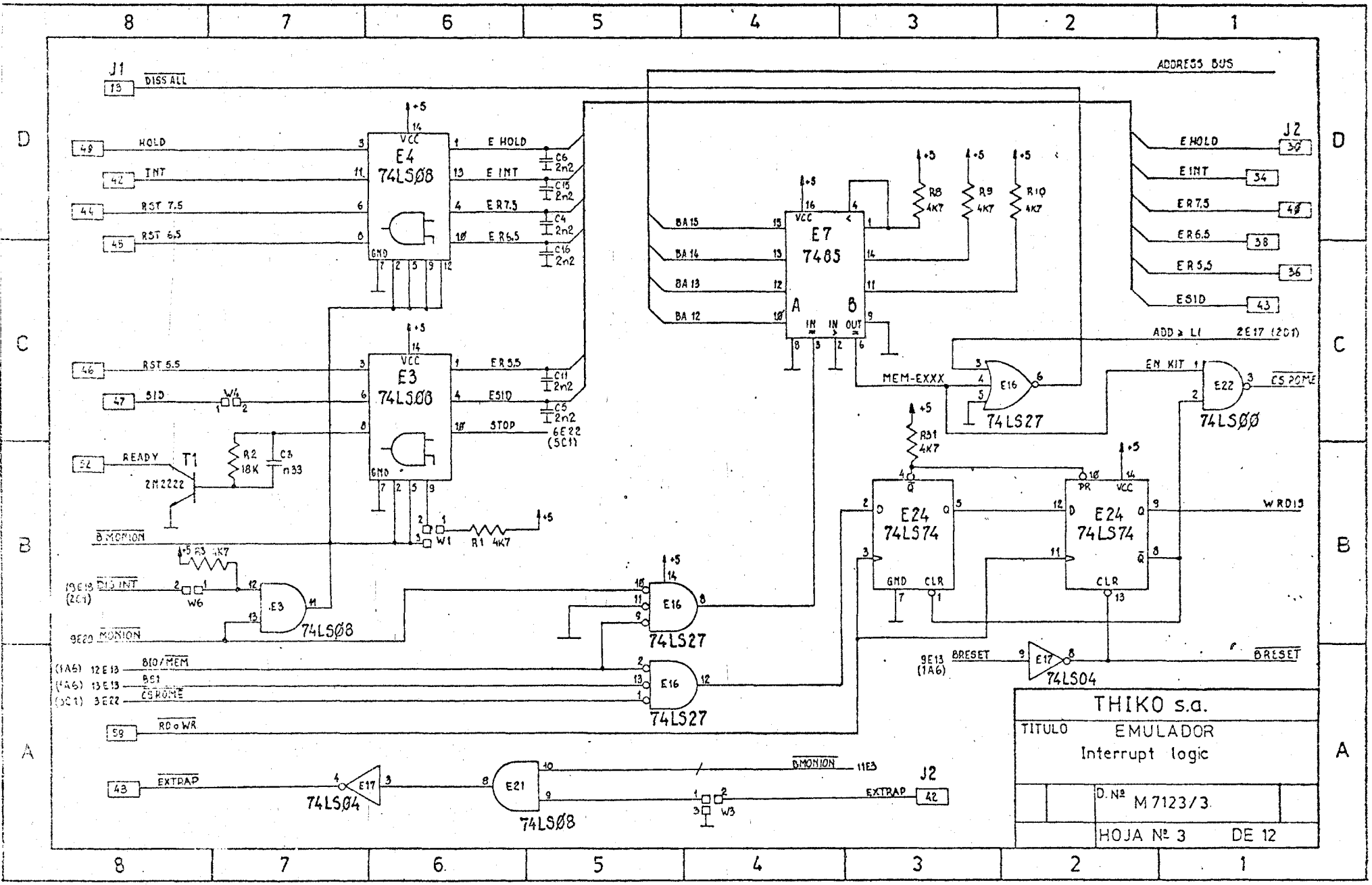
Ahora estamos en condiciones de examinar el programa y sus subrutinas instruccion por instruccion o colocando break-points por software (estas dos posibilidades se realizan con los comandos del programa Monitor), o sino mediante el SINGLE-CYCLE DISPLAY BOARD, colocando break-points por hardware en cualquier lugar de memoria o de I/O o siguiendo las instrucciones ciclo a ciclo de maquina para detectar posibles errores de hardware.

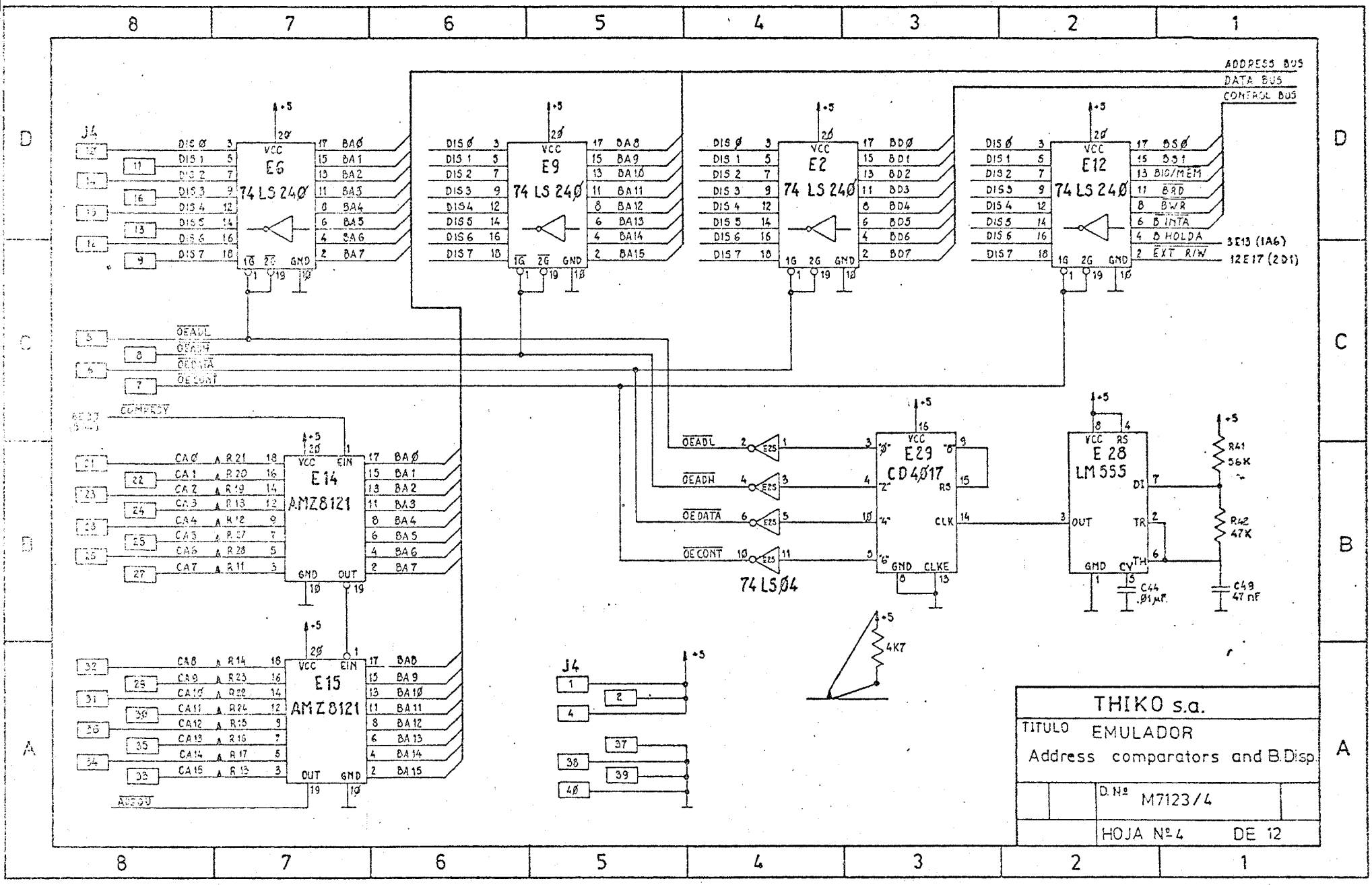
Mediante el empleo de los comandos del monitor podemos alterar los datos del programa (puesto que ahora estan en RAM) y modificar las rutinas hasta que cumplan con los requisitos del programa deseado.

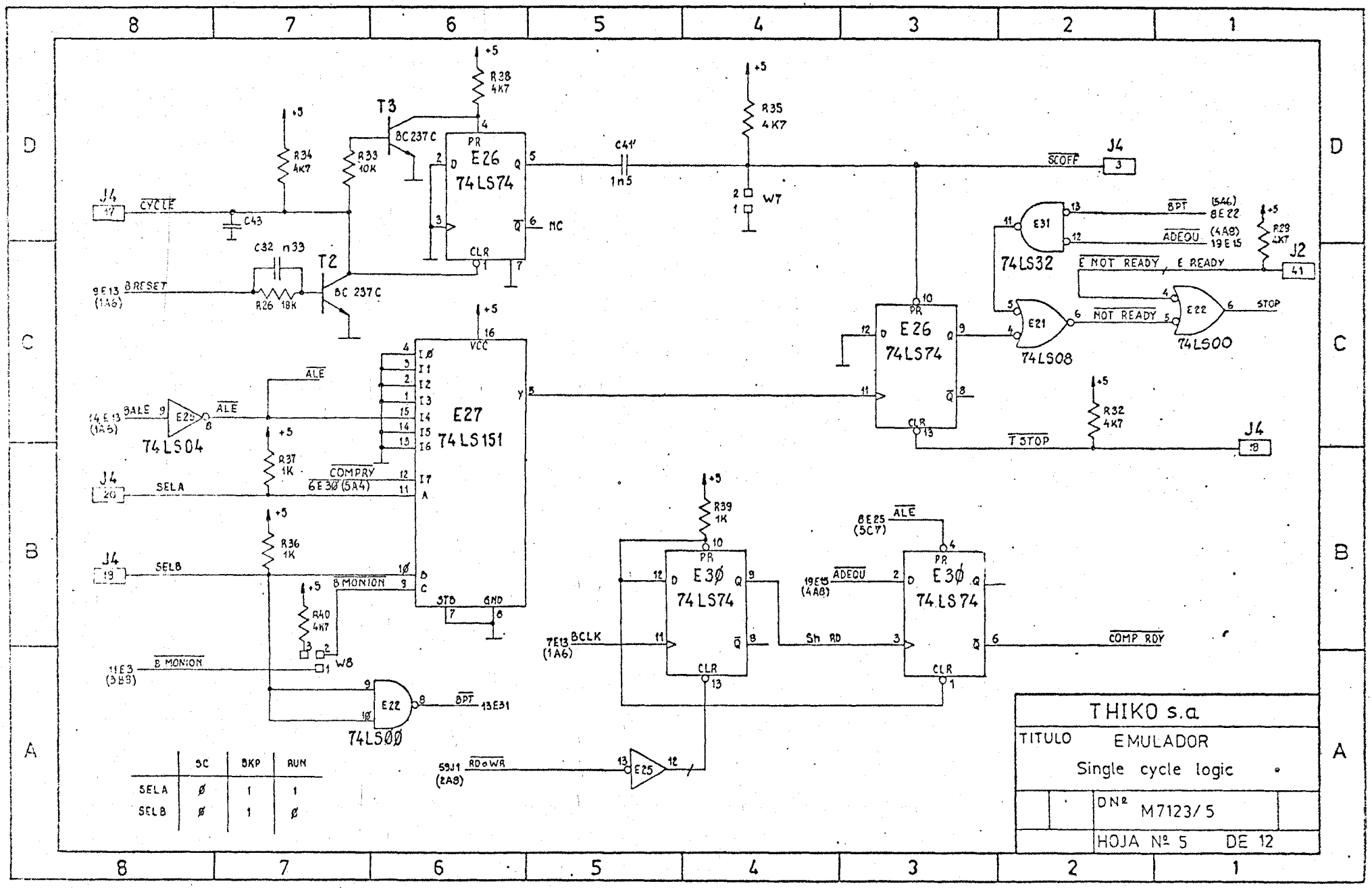
Una vez que el nuevo programa quedo definitivamente modificado, el siguiente paso es salvar los datos que tenemos en RAM en una Eprom o en un archivo de cassette. Para realizar cualquiera de estas dos posibilidades se deben utilizar los comandos del Prom Programmer o la rutina de Write Cassette del kit.


DATA BUS
ADDRESS BUS
CONTROL BUS
E1 74LS245
E5 74LS244
E8 74LS244
E13 74LS244
E11 1/2 74LS244
J1
J2
BD Ø
BD 1
BD 2
BD 3
BD 4
BD 5
BD 6
BD 7
BA Ø
BA 1
BA 2
BA 3
BA 4
BA 5
BA 6
BA 7
BA 8
BA 9
BA 10
BA 11
BA 12
BA 13
BA 14
BA 15
BALE
BS Ø
BS 1
BIO/MEM
BHOLDA
BRESET
BCLK
BSOD
B HOLDA
ERRØ
READ
WRITE
INTA
ERD
EWR
EINTA
EIO/MEM
EXR/W
ALE
Gnd
R4 4K7
R5 4K7
R6 4K7
R7 4K7
W2
W5
L1
R30 180Ω
C24 .039 µF OPCIONAL
▲ Ver nota
▲ DIR
Ø B→A
1 A→B
THIKO s.a.
TITULO EMULADOR
Buffers: Address Data Control
Control
Fecha
21.3.85
D.Nº M7123/1
HOJA Nº 1 DE 12


DATA BUS
ADDRESS BUS
ADD > LT
EXT R/W
B HOLDA
DIOINT
E10
AM 2732
CE ROM
BRD
DISS ALL
ENRD
E21
74LS08
R25
4K7
J1
MONITOR-ON
RD o WR
E20
74 LS 74
MONION
MONIOFF
BRESET
E31
74LS32
E17
74LS04
E23
7485
E19
7485
WRDIS
MONION
E18
74 LS374
E22
74LS00
MEM-EXXX
BWR
THIKO s.a.
TITULO EMULADOR
Limit Logic and Expansion ROM
D.Nº M7123 / 2
HOJA Nº 2 DE 12


J1
DISS ALL
ADDRESS BUS
HOLD
INT
RST 7.5
RST 6.5
RST 5.5
SID
READY
E4
74LS08
E3
74LS08
E HOLD
E INT
E R7.5
E R6.5
E R5.5
ESID
STOP
6E22 (5C11)
C6 2n2
C15 2n2
C4 2n2
C16 2n2
C11 2n2
C5 2n2
W4
T1
2N2222
R2 18K
C3 n33
W1
R1 4K7
B MONION
R3 4K7
W6
E3
74LS08
E7
7485
BA 15
BA 14
BA 13
BA 12
R8 4K7
R9 4K7
R10 4K7
J2
E HOLD
E INT
E R7.5
E R6.5
E R5.5
ESID
ADD ≥ L1
2E17 (2D1)
MEM-EXXX
E16
74LS27
EN KIT
E22
74LS00
CS ROME
R31 4K7
E24
74LS74
E24
74LS74
WRD13
E16
74LS27
E16
74LS27
BRESET
E17
74LS04
BRESET
BIO/MEM
BS1
CS ROME
RD o WR
EXTRAP
E17
74LS04
E21
74LS08
B MONION
EXTRAP
W3
THIKO s.a.
TITULO
EMULADOR
Interrupt logic
D. Nº M 7123/3
HOJA Nº 3 DE 12


ADDRESS BUS
DATA BUS
CONTROL BUS
E6 74 LS 240
E9 74 LS 240
E2 74 LS 240
E12 74 LS 240
E14 AMZ8121
E15 AMZ8121
E29 CD4017
E28 LM555
74 LS04
J4
OEADL
OEADH
OEDATA
OECONT
COMPEQY
ADEQU
BIO/MEM
BRD
BWR
B INTA
B HOLDA
EXT R/W
3E13 (1A6)
12E17 (2D1)
R41 56K
R42 47K
C44 .01µF
C49 47 nF
4K7
THIKO s.a.
TITULO EMULADOR
Address comparators and B.Disp.
D.Nº M7123/4
HOJA Nº 4 DE 12



	SC	BKP	RUN
SELA	Ø	1	1
SELB	Ø	1	Ø

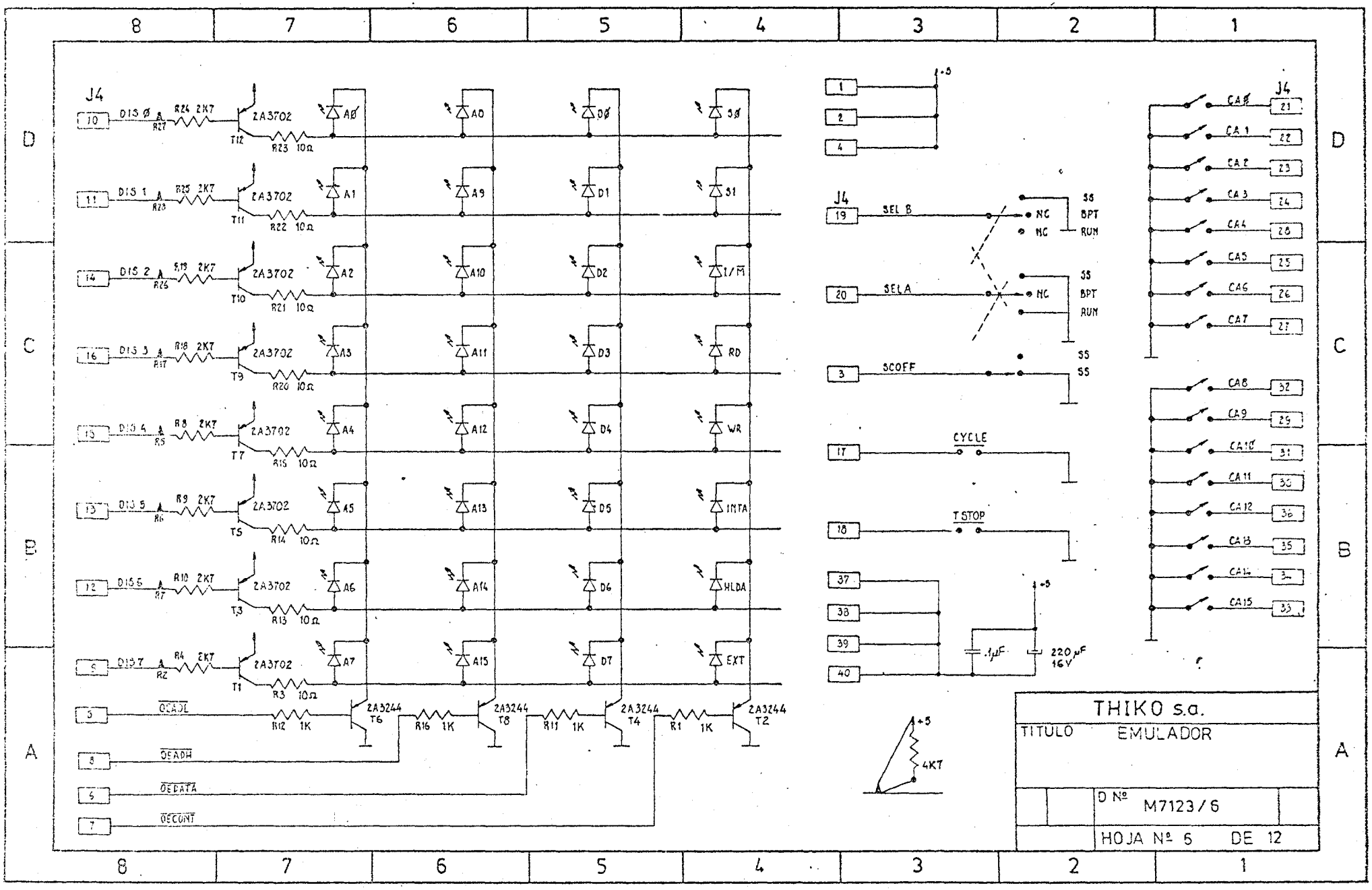
THIKO s.a.

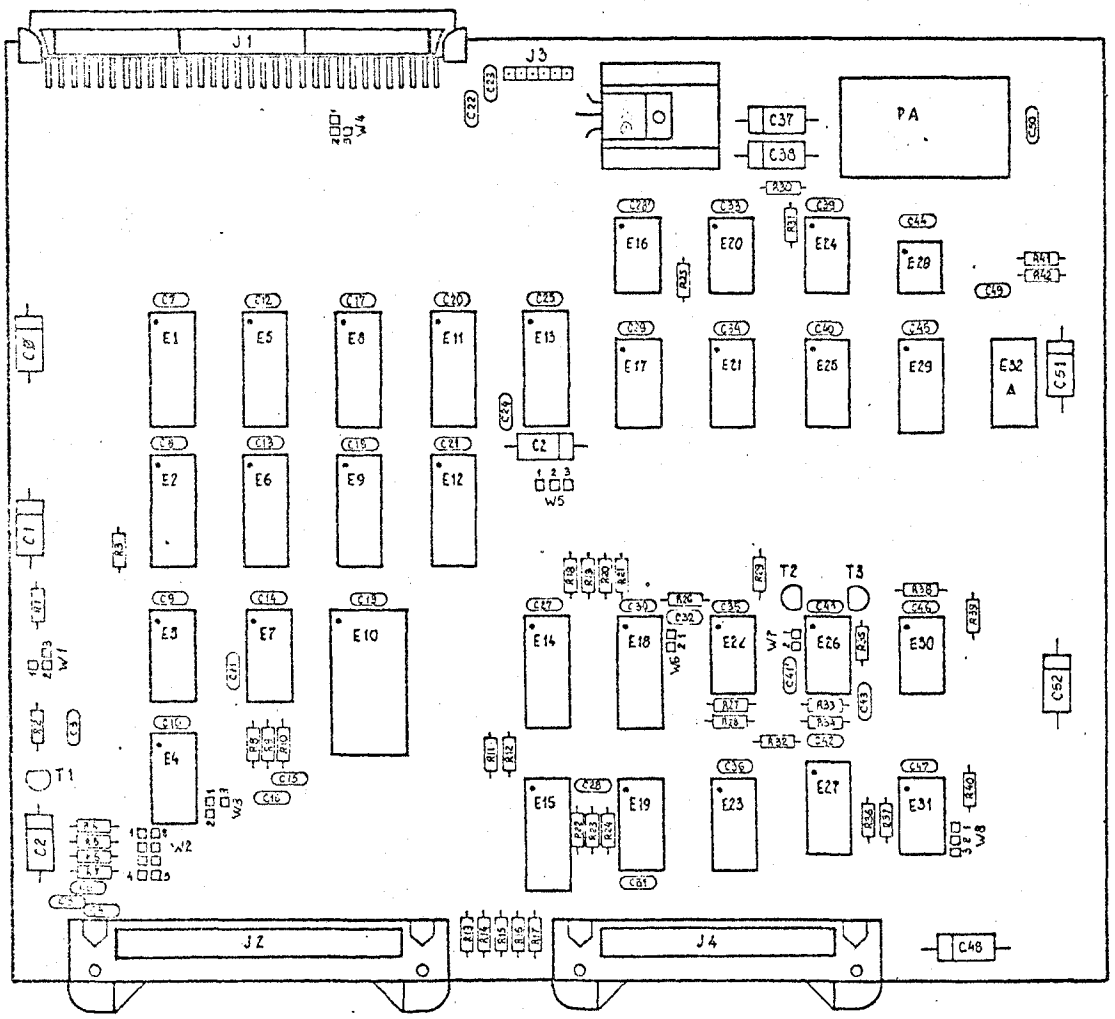
TITULO EMULADOR

Single cycle logic

DNº M7123/5

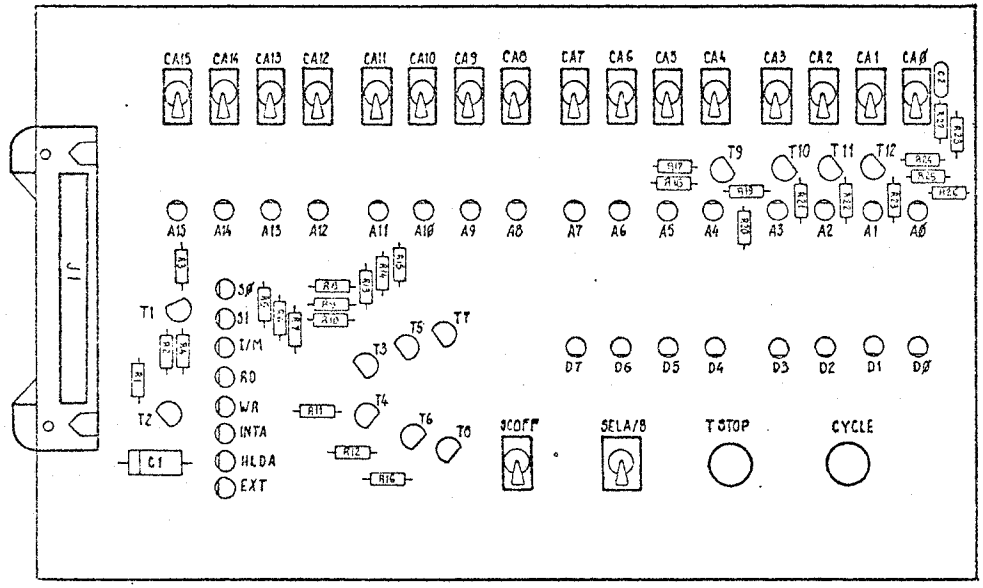
HOJA Nº 5 DE 12


J4
DIS Ø
R24 2K7
R27
2A3702
T12
R23 10Ω
AØ
AØ
DØ
SØ
DIS 1
R25 2K7
R28
T11
R22 10Ω
A1
A9
D1
S1
DIS 2
R26 2K7
T10
R21 10Ω
A2
A10
D2
I/M
DIS 3
R18 2K7
R17
T9
R20 10Ω
A3
A11
D3
RD
DIS 4
R8 2K7
R5
T7
R15 10Ω
A4
A12
D4
WR
DIS 5
R9 2K7
R6
T5
R14 10Ω
A5
A13
D5
INTA
DIS 6
R10 2K7
R7
T3
R13 10Ω
A6
A14
D6
HLDA
DIS 7
R4 2K7
R2
T1
R3 10Ω
A7
A15
D7
EXT
OCADL
R12 1K
2A3244
T6
R16 1K
2A3244
T8
R11 1K
2A3244
T4
R1 1K
2A3244
T2
OEADH
OEDATA
OECONT
+5
SEL B
NC
NC
SS
BPT
RUN
SELA
SCOFF
CYCLE
T STOP
.1μF
220μF
16V
4K7
CAØ
CA 1
CA 2
CA 3
CA4
CA5
CA6
CA7
CA8
CA9
CA10
CA 11
CA 12
CA 13
CA14
CA15
THIKO s.a.
TITULO EMULADOR
D Nº M7123/6
HOJA Nº 6 DE 12



▲ Disponible

THIKO s.a.	
TITULO EMULADOR	
Lay Out	
D.Nº M7123/7	
HOJA Nº 7	DE 12



THIKO s.a.	
TITULO	EMULADOR Lay out Single Cycle
D.Nº	M7123/8
HOJA Nº 8	DE 12

Listado del Conector J1 del N3

1	USER Ø	2	GND	33	$\overline{\text{READ}}$	34	$\overline{\text{WRITE}}$
3	USER 1	4	GND	35	IO/$\overline{\text{MEM}}$	36	ALE
5	DØ	6	D1	37	SØ	38	S1
7	D2	8	D3	39	USER 3	40	RESET
9	D4	1Ø	D5	41	$\overline{\text{INTA}}$	42	INT
11	D6	12	D7	43	$\overline{\text{EXTRAP}}$	44	RST 7,5
13	$\overline{\text{DIS ALL}}$	14	USER 2	45	RST 6,5	46	RST 5,5
15	AØ	16	A1	47	SID	48	SOD
17	A2	18	A3	49	HOLD	50	HOLDA
19	A4	20	A5	51	CLK OUT	52	READY
21	A6	22	A7	53	$\overline{\text{ZONE Ø}}$	54	$\overline{\text{ZONE 1}}$
23	A8	24	A9	55	$\overline{\text{ZONE 2}}$	56	$\overline{\text{ZONE 3}}$
25	A1Ø	26	A11	57	$\overline{\text{MONION}}$	58	USER 6
27	A12	28	A13	59	$\overline{\text{RD o WR}}$	60	KIT SEL
29	A14	30	A15	61	USER 4	62	GND.
31	$\overline{\text{RAM SEL}}$	32	$\overline{\text{FIRST 8K}}$	63	USER 5	64	GND

THIKO s.a.

TITULO EMULADOR
Listado del Conector J1

D Nº M7123/9

HOJA Nº 9 DE 12

Listado del Conector J2 del N3

1	Gnd	2	Gnd
3	BD 1	4	BDØ
5	BD 3	6	BD2
7	BD 5	8	BD4
9	BD 7	10	BD6
11	BA 9	12	BA8
13	BA8	14	BA10
15	BA 13	16	BA 12
17	BA 15	18	BA 14
19		20	
21		22	
23		24	
25	$\overline{EWR}$	26	$\overline{ERD}$
27		28	
29	B RESET	30	E HOLD
31	BSØ	32	$\overline{E\ INTA}$
33	BALE	34	E INT
35		36	E R 5,5
37	BS1	38	ER6,5
39	E IO/$\overline{MEM}$	40	ER 7,5
41	E READY	42	EXTRP
43	E S1D	44	B CLK
45	B S0D	46	B HOLDA
47	Gnd	48	Gnd
49	Gnd	50	Gnd

J3

1	Vin
2	NC
3	Gnd
4	Gnd
5	NC
6	Vin

THIKO s.a.

TITULO EMULADOR

Listado de Conectores J2 y J3

D Nº M7123/10

HOJA Nº 10 DE 12

Listado del Conector J4 del N3

1	+5	2	+5
3	SCOFF	4	+5
5	OEADL	6	OEDATA
7	OECONT	8	OEADH
9	DIS 7	10	DIS 0
11	DIS 1	12	DIS 6
13	DIS 5	14	DIS 2
15	DIS 4	16	DIS 3
17	CYCLE	18	TSTOP
19	SELA	20	SELB
21	CA Ø	22	CA 1
23	CA 2	24	CA 3
25	CA 5	26	CA 6
27	CA 7	28	CA 4
29	CA 9	30	CA11
31	CA 10	32	CA 8
33	CA 15	34	CA14
35	CA 13	36	CA12
37	Gnd.	38	Gnd.
39	Gnd.	40	Gnd.

THIKO s.a.

TITULO EMULADOR

Listado del Conector J4

D.Nº M7123/ 11

HOJA Nº11 DE 12

DENOM.	VALOR
R1	4K7
R2	18K
R3	4K7
R4	4K7
R5	4K7
R6	4K7
R7	4K7
R8	4K7
R9	4K7
R10	4K7
R11	4K7
R12	4K7
R13	4K7
R14	4K7
R15	4K7
R16	4K7
R17	4K7
R18	4K7
R19	4K7
R20	4K7
R21	4K7
R22	4K7
R23	4K7
R24	4K7
R25	4K7
R26	18K
R27	4K7
R28	4K7
R29	4K7
R30	180Ω
R31	4K7
R32	4K7
R33	4K7
R33	10K
R34	4K7
R35	4K7
R36	1K
R37	1K
R38	4K7
R39	1K
R40	4K7
R41	56K
R42	47K

DENOM.	VALOR	DENOM.	VALOR
C0	47µF 16V	C28'	.1µF
C1	47µF 16V	C29	.1µF
C2	47µF 16V	C30	.1µF
C3	n33	C31	.1µF
C4	2n2	C32	n33
C5	2n2	C33	.1µF
C6	2n2	C34	.1µF
C7	.1µF	C35	.1µF
C8	.1µF	C36	.1µF
C9	.1µF	C37	47µF 16V
C10	.1µF	C38	47µF 16V
C11	2n2	C39	.1µF
C12	.1µF	C40	.1µF
C13	.1µF	C41	.1µF
C14	.1µF	C41'	1n5
C15	2n2	C42	.1µF
C16	2n2	C43	.01µF
C17	.1µF	C44	.1µF
C18	.1µF	C45	.1µF
C19	.1µF	C46	.1µF
C20	.1µF	C47	.1µF
C21	.1µF	C48	47µF 16V
C22	.1µF	C49	47n
C23	.1µF	C50	.1µF
C24	39n	C51	47µF 16V
C25	.1µF	C52	47µF 16V
C26	47µF 16V		
C27	.1µF		
C28	n33		

DENOM.	VALOR
T1	2N2222
T2	BC237C
T3	BC237C
L1	CQY40

DENOM.	VALOR
E1	74LS245
E2	74LS240
E3	74LS08
E4	74LS08
E5	74LS244
E6	74LS240
E7	7485
E8	74LS244
E9	74LS240
E10	2732
E11	74LS244
E12	74LS240
E13	74LS244
E14	8121
E15	8121
E16	74LS27
E17	74LS04
E18	74LS374
E19	7485
E20	74LS74
E21	74LS08
E22	74LS00
E23	7485
E24	74LS74
E25	74LS04
E26	74LS74
E27	74LS151
E28	LM555
E29	CD4017
E30	74LS74
E31	74LS32
E32	LIBRE
—	LM340T5

THIKO s.a.

TITULO
EMULADOR
Lista de Componentes

D.Nº M7123/12

HOJA Nº 12 DE 12